WHAT IS CLAIMED IS:

1. A system for securing an application for execution on a computer, the system comprising:

a server computer;

a network; and

a client computer operably connected to the server computer via the network;

wherein the client computer receives an application from the server computer;

wherein the client computer executes the application subsequent to receiving the application; and

wherein the client computer includes an interception module for intercepting at least one network request from the application, wherein the interception module determines whether the destination address is listed in a set of approved addresses, and wherein the interception module notifies a proxy that the request is intercepted.

2. A method of securing an application for execution on a computer, the method comprising:

modifying a binary of the application such that a request from the application to transmit data over the network is intercepted, wherein the request identifies a destination address; and

determining whether the destination address is listed in a set of approved addresses.

3. The method of Claim 2, additionally comprising notifying a proxy that the request is intercepted.

4. The method of Claim 2, additionally comprising inserting in an import table a reference to an interception module, wherein the reference is inserted in the import table such that the interception module is invoked in response to loading of the application, and wherein the interception module intercepts the request from the application.

5. The method of Claim 2, wherein the network request is an network accept request and wherein the method additionally comprises:

determining whether there is an entry in a connection queue;

if there is no entry in the connection queue, blocking until there is an entry in the connection queue.

6. The method of Claim 2, wherein the network request is an network accept request and wherein the method additionally comprises:

determining whether there is an entry in a connection queue;

if there is no entry in the connection queue, returning a message indicating that there is no entry in the connection queue.

7. The method of Claim 2, wherein the network request is a network send request and wherein the method additionally comprises writing the content of a buffer that is provided by the application into a send queue.

8. The method of Claim 2, wherein the network request is a network receive request and wherein the method additionally comprises reading the contents of a buffer that is in a proxy table and returning the contents of the buffer to the application.

9. The method of Claim 2, wherein the network request is a socket request and wherein the method additionally comprises recording the socket request and transmitting to the application a unique socket identifier.

10. The method of Claim 2, wherein the network request is a network bind request and wherein the method additionally comprises obfuscating the network address.

11. The method of Claim 2, wherein the network request is a network connect request and wherein the method additionally comprises updating a status flag indicating that a socket that is identified by the network is virtually connected to a selected destination socket.

12. The method of Claim 2, wherein the network request is a network listen request and wherein the method additionally comprises updating a status flag indicating that a socket is listening for communications from remote destinations.

13. The method of Claim 4, additionally comprising encrypting the data.

14. A system for securing an application for execution on a computer, the system comprising:

means for modifying a binary of the application such that a request from the application to transmit data over the network is intercepted, wherein the request identifies a destination address; and

means for determining whether the destination address is listed in a set of approved addresses.

15. The system of Claim 14, additionally comprising means for notifying a proxy that the request is intercepted.

16. The system of Claim 14, wherein the network request is an network accept request and wherein the method additionally comprises:

means for determining whether there is an entry in a connection queue; and

means for if there is no entry in the connection queue, blocking until there is an entry in the connection queue.

17. The system of Claim 14, wherein the network request is a network send request and wherein the method additionally comprises writing the content of a buffer that is provided by the application into a send queue.

18. The method of Claim 14, wherein the network request is a network receive request and wherein the method additionally comprises writing the contents of a buffer that is that is in a proxy table and returning the contents to the application.

19. The system of Claim 14, wherein the network request is a socket request and wherein the method additionally comprises recording the socket request and transmitting to the application a unique socket identifier.

20. The system of Claim 14, wherein the network request is a network bind request and wherein the method additionally comprises obfuscating the network address.

21. The system of Claim 14, wherein the network request is a network connect request and wherein the method additionally comprises updating a status flag indicating that a socket that is identified by the network is virtually connected to a selected destination socket.

22. The system of Claim 14, wherein the network request is a network listen request and wherein the method additionally comprises updating a status flag indicating that a socket is listening for communications from remote destinations.

23. The system of Claim 14, additionally comprising means for encrypting the data.

24. A system for securing an application for execution on a computer, the system comprising:

a preprocessor module for modifying the binary of an application such that a request to transmit data over the network is intercepted, wherein the request identifies a destination address, wherein the interception module determines whether the destination address is listed in a set of approved addresses, and wherein the interception module notifies a proxy that the request is intercepted.